Introduction to Linux-based solution for embedded software development

Section 1 Eddy Real-Time Linux, Lemonix

Section 2 Eddy Integrated Development Environment, LemonIDE

Section 3 Eddy Utility Program

Introduction to Eddy Utility Program

SystemBase's self-developed utility programs COM Port Redirector, PortView and TestView are designed for fluent performance of Portbase, Eddy and Wicomm. The COM Port Redirector which can COM-communicate by creating a virtual COM-Port on the controlled PC can still be used on former control programs controlled by COM-Port. PortView, designed for remote monitoring device products, can monitor components needed for function such as status information, serial port status and serial data's debugging. TestView, designed for testing each serial communication products, has wider range of testing communication with its ability to communicate in TCP/UDP Server-Client mode as well as COM communication. SNMP is also provided enabling an easy monitor and management of devices connected to a network.

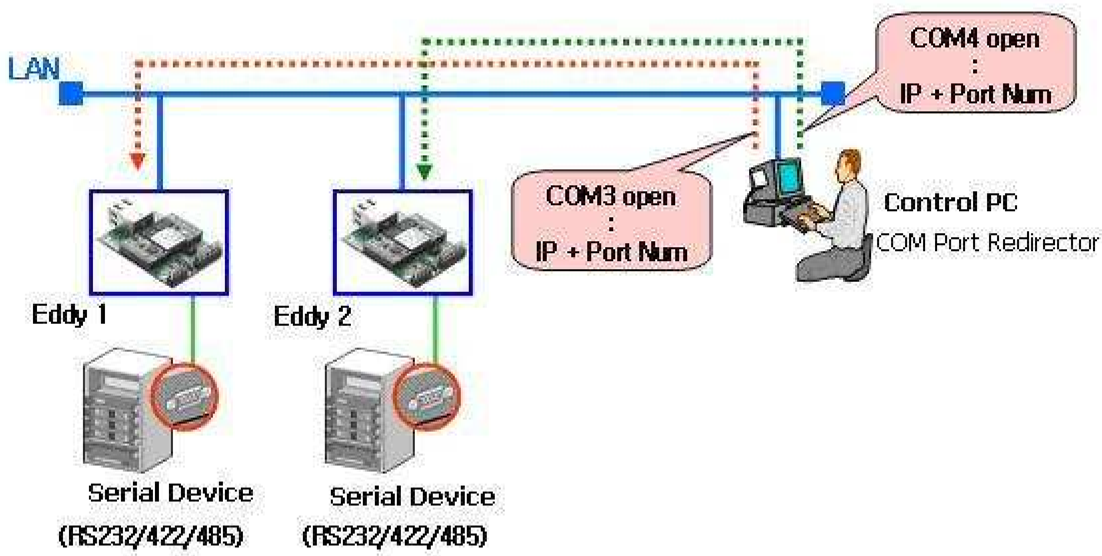
Next will be special features of each program.

Virtual COM Port Program

COM Port Redirector

1. Summary

COM Port Redirector is a network COM port driver and Redirector Control which operates it. It allows the device server's serial ports to be used as local COM ports of PC. Up to 255 COM ports can be registered on one PC using COM Port Redirector, allowing the serial ports connected to device server to be used as if they were COM ports connected on the user's PC.

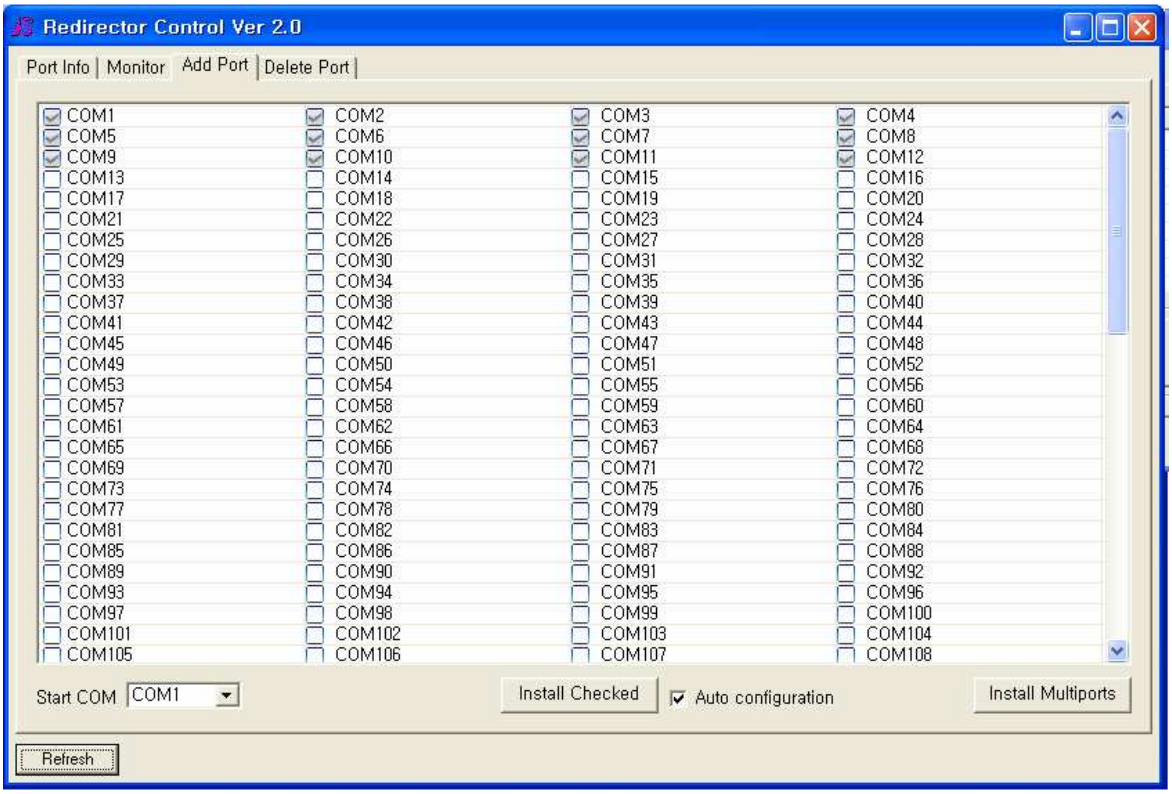


[* Understanding COM Port Redirector]

Therefore, users can connect to Eddy on LAN just by opening COM port with COM Port Redirector on the controlling PC which makes remote control possible.

2. How to Use

Screen Components



COM Port Redirector is composed of Port Info, Monitor, Add Port and Delete Port.

[* COM Port Redirector Screen]

Port Info : Port setting reference & modification

Monitor : Port status reference

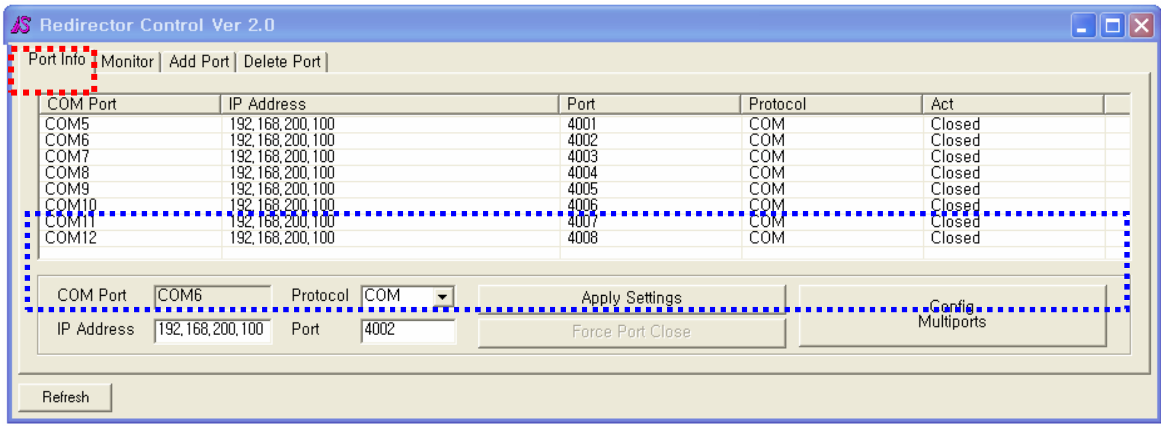
Add Port : Port install

Delete Port : Port uninstall

2.1 Port Info

Referencing and modifying Redirector COM port settings are done on “Port Info” tab.

As shown below, Redirector COM ports are all displayed and each port’s setting and status are shown.



Port Info Abilities

COM Port : Virtual COM port device name

IP Address : Eddy’s IP address the virtual COM port is going to connect

Port : Eddy’s port number the virtual COM port is going to connect

Protocol : Communication method with Eddy (COM/Encryption/Raw)

. COM : General virtual COM Port (default)

. Encryption : Coded communication with device server

(Uses Korean standard symmetric key way 128bit block coding algorithm SEED)

. Raw : General virtual COM mode sending only pure data

Act : Shows status of COM Ports (Closed/Open)

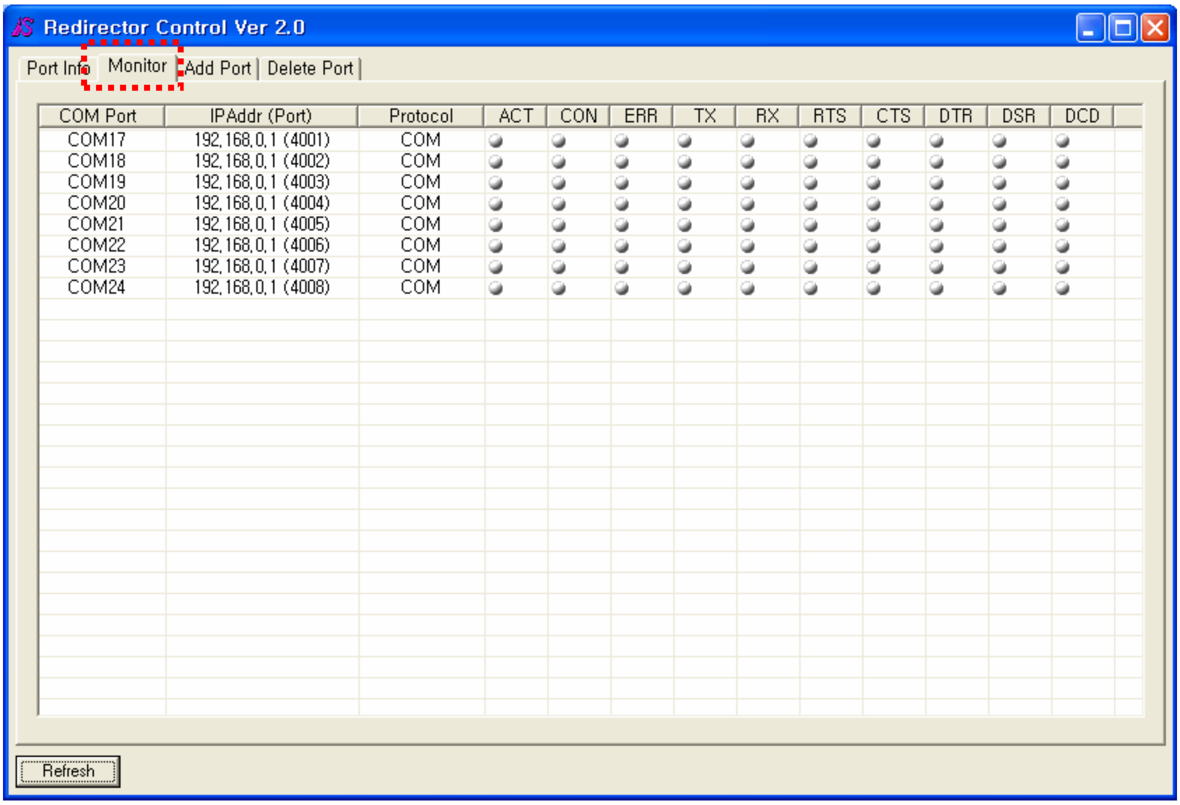
Apply Settings : Applies modified values.

Force Port Close : Used to force close opened ports when they do not close properly.

Config Multiports : Used to apply modified values to many ports on one time.

2.2 Reference to Port Status(Monitor)

Use “Monitor” tab to view current status of Redirector COM ports..



Each columns shows following information

ACT : Turns blue when application program is using the COM port.

CON : Turns blue when the COM port is connected to the device server.

ERR : Turns red when there is a connection problem with Eddy device.

Tx : Turns green when data is being transmitted

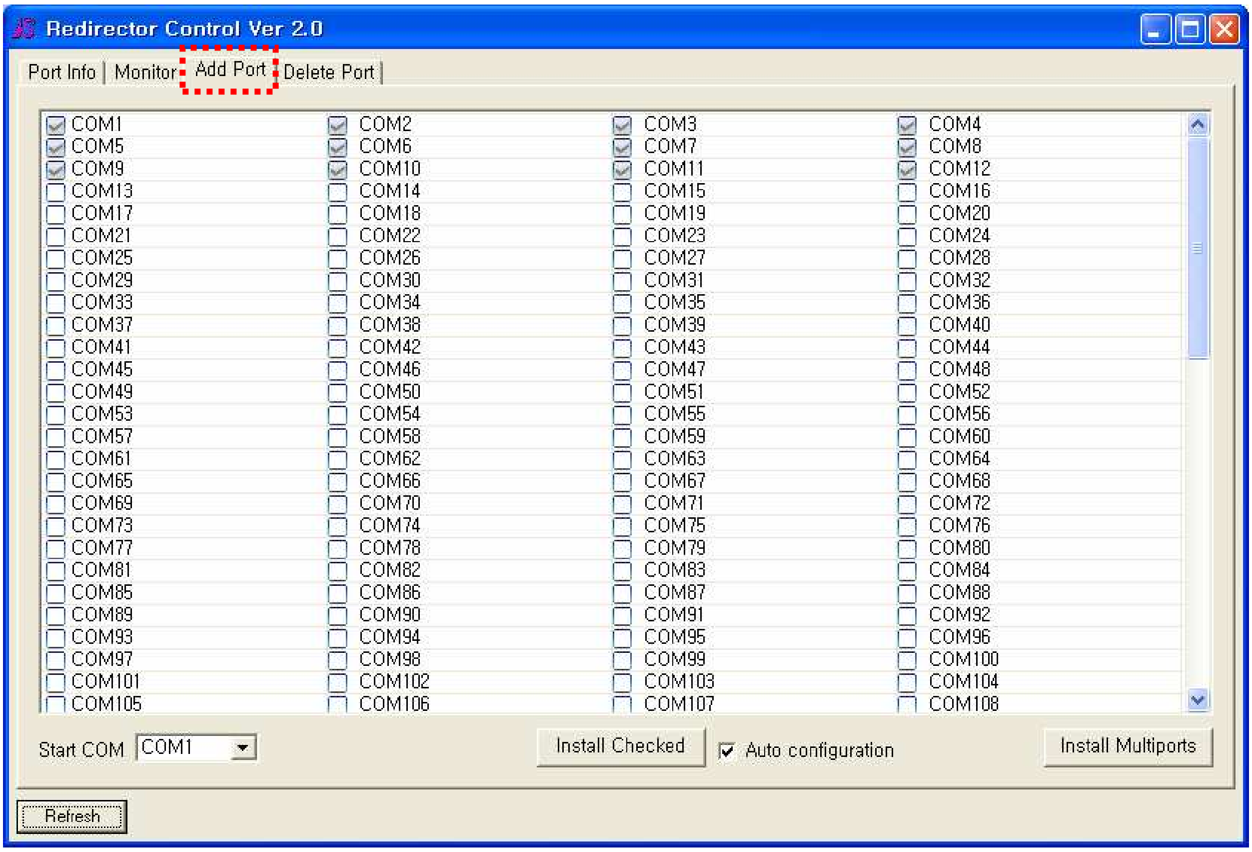
Rx : Turns yellow when data is being received.

RTS, DCD : Shows COM port status. Turns blue.

Refresh Button: When COM port number is modified by some other way other than through COM Port Redirector, or other serial port device is installed, clicking “Refresh” button or reactivating the tab will update COM Port Redirector.

2.3 Port Installation(Add Port)

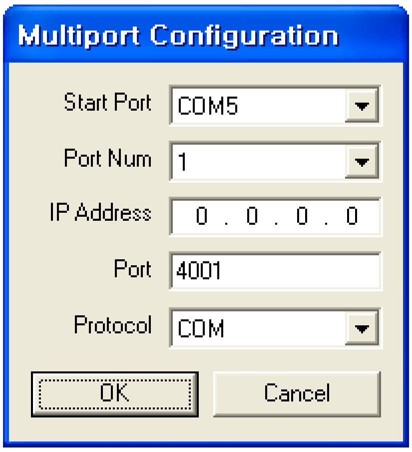
Adding Redirector COM port is done on “Add Port” tab.



Users can check and install desired COM port number for checkable numbers.

Numbers 1 through 255 are shown. To select other COM port numbers you can specify the starting COM port number in combo box in “Start COM”.

Install Multiport button : Used to line up COM port numbers.



- Usable COM port numbers are listed in “Start Port”. The number selected on “Port Num” is the amount of ports installed starting from the port number selected on “Start Port”.

- COM port number increases by one, and if a number cannot be used, next number is used.

Example) Start Port = COM5, Port Num = 4 Selected.

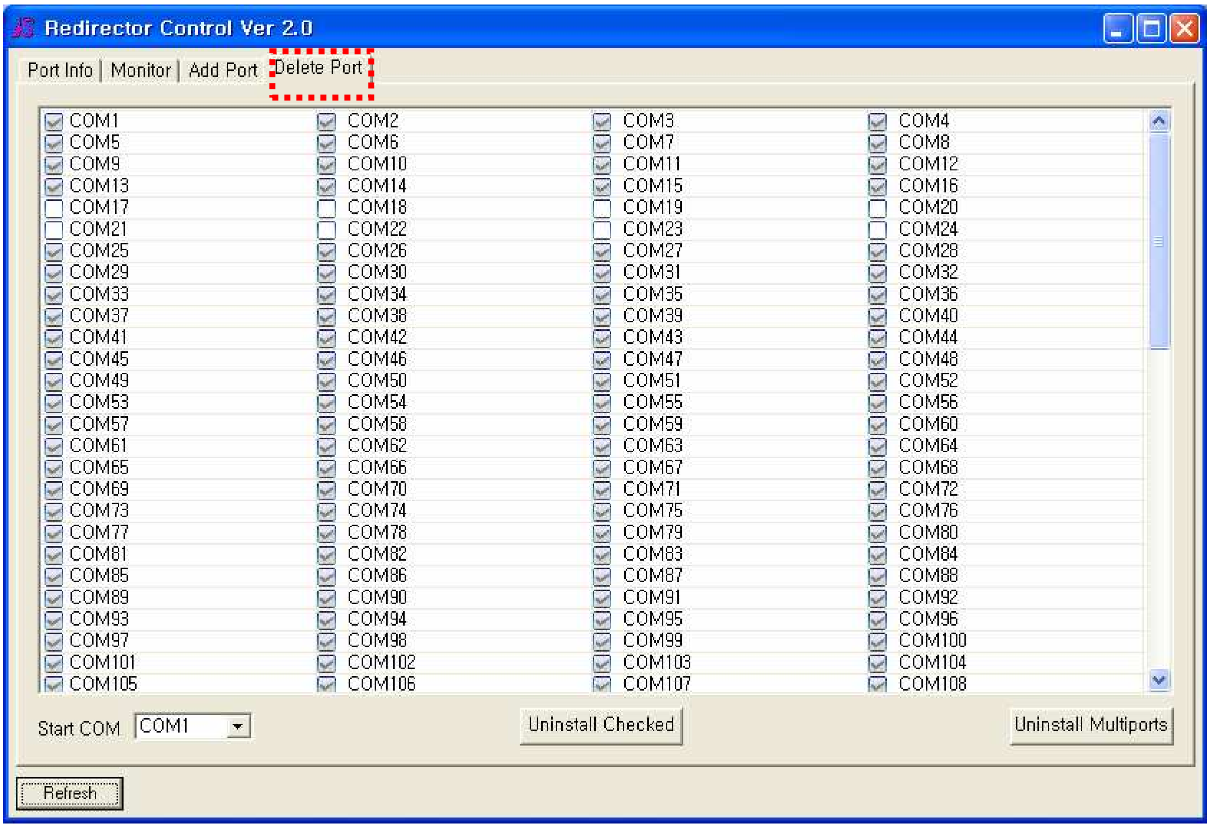
COM6, COM8 unusable.

→ COM5, COM7, COM9, COM10 installed.

2.3 Port Removal

Redirector COM port removal is done on “Delete Port” tab.

Currently installed Redirector COM ports are checkable.



256 ports are shown from COM1. To select other COM port numbers you can specify the starting COM port number in combo box in “Start Com”.

Port is removed after checking the port you wish to remove and clicking “Uninstall Checked”. Click “Uninstall Multiports” if you wish to remove all ports.

2.4 Using Ports

The installed virtual COM ports can be used with the same way as general serial ports.

They can be used with the same way as general serial ports on general console application programs such as hyper terminal, SecureCRT and users can program applications programs that use COM ports using Windows

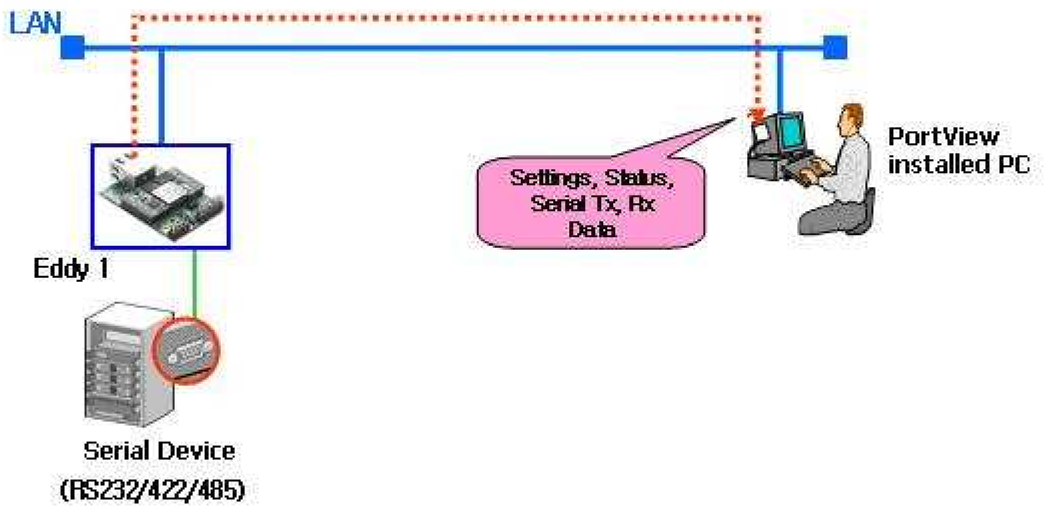
API.
COM Port Redirector must be running in order to use Redirector COM ports and COM port settings should be checked before they are used on application programs.

Eddy Serial Monitoring Program

PortVew

1. Summary

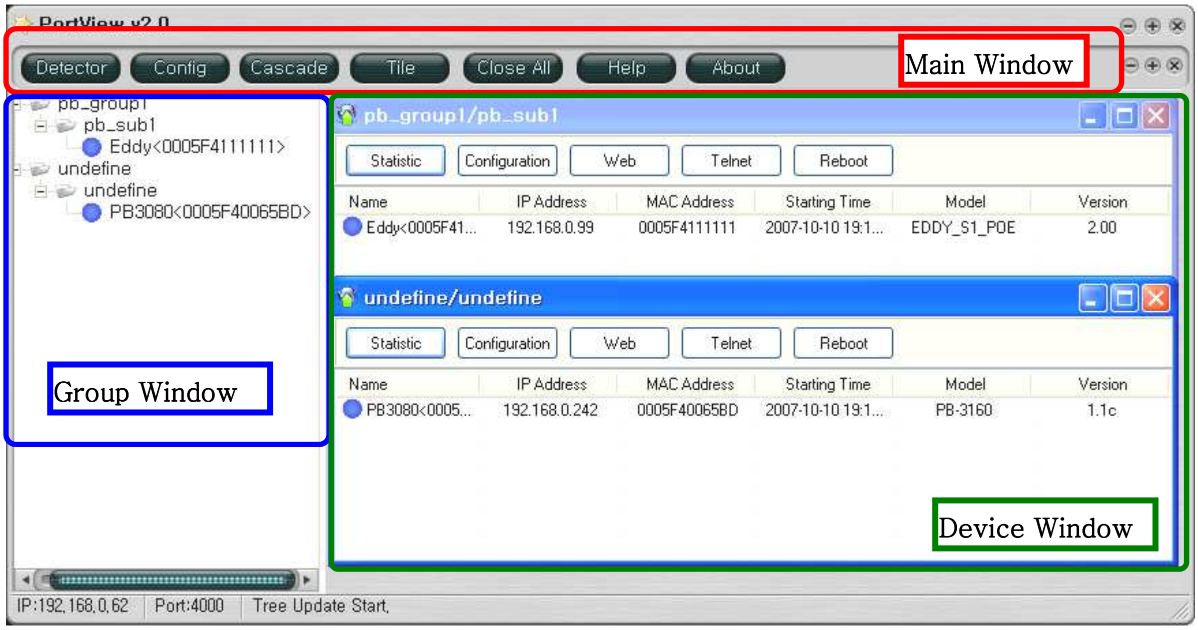
PortView is a application program for Windows that can monitor device server's serial port status and function on remote PC.



[Understanding PortView]
PortView Primary Functions
Detector Ability (Eddy 의 MAC Search)
Real-time data monitoring Ability(Scope Ability)
Real-time device status reference ability(Config Ability)
Group setting and managing ability
Web, Telnet quick immediate connection ability
Therefore, PortView Ver2.0, unlike former versions, has many abilities such as Detector ability which searches Eddy's MAC on initial connection and quick connection to Web and Telnet.

2. How to Use

Screen Components



PortView consists of 3 parts as shown below.

Main Window

Shown by red line. Manages PortView.

Detector: Detects device servers in local network.

Config: Configure Alarm, Log, Service Socket, Password.

Cascade: Show Device Windows in PortView(Cascade)

Tile: Show Device Windows in PortView.

Close All: Close all Device Windows.

Help: Opens folder containing PortView menu.

About: Shows program version.

Group Window

Shown by blue line. Shows group, subgroup, device server and undefined.

Group: The upper part of folder tree on the picture. Groups can be created and deleted as users will.

Ex) “pb_group1”: renamed.

Subgroup: Can be created inside group. Contains device list.

Device: Shown inside group. First appears on “undefined” subgroup and can be moved to another subgroup by left click and dragging..

Device Window

Shown by green line. Show device lists by subgroups. There are two shown on above picture. Names can be

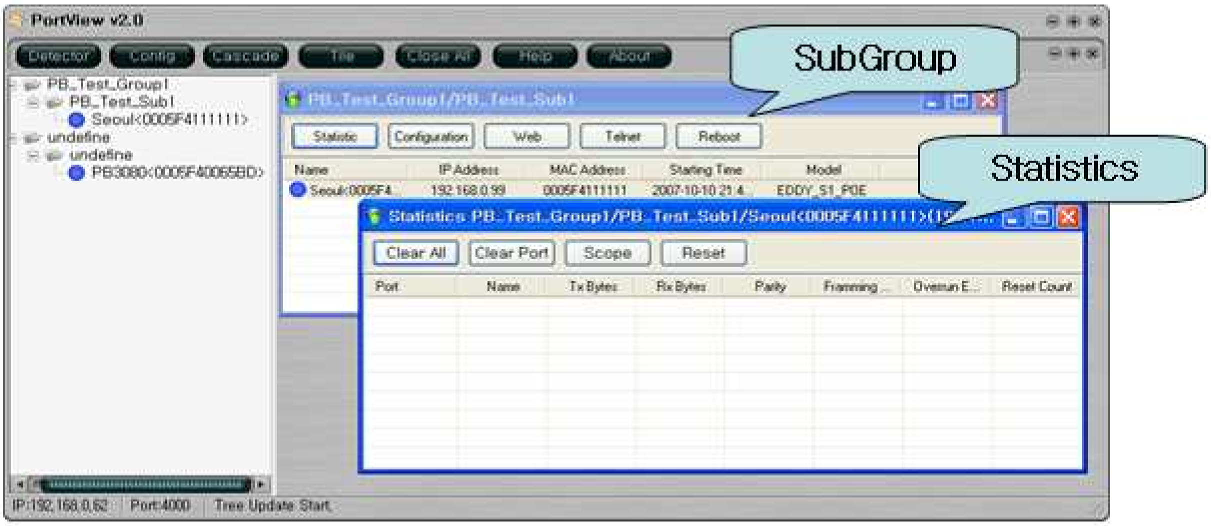
changed except for “Undefined”, devices can be moved from one subgroup to another with mouse dragging.

It is where management on each device’s information, status, data IO scope ability actually takes place and device monitoring, primary function of PortView, is done.

Device Window is divided into two groups.

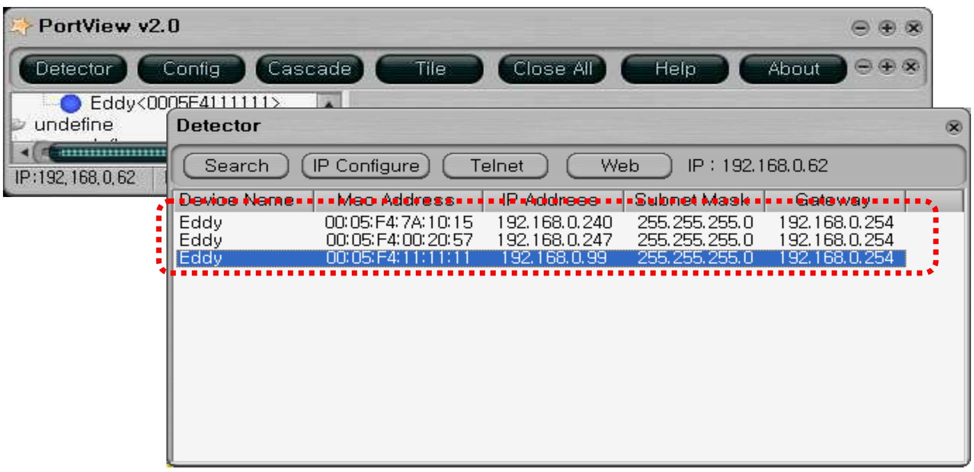
SubGroup : “PB_Test_Group/PB_Test_Sub1”window in the picture below. Name is created by adding the group it belongs and the subgroup. Shows list of devices contained in the subgroup.

Statistics : Named as “Statistics GroupName/SubGroupName/DeviceName”. Monitors device ports. Provides Tx/Rx, Scope ability.



2.1 Detector

Searches devices that can be managed by PortView contained in the Local Area Network. The detector searches Eddy’s MAC Address.



Search: Detects devices in the local area. .

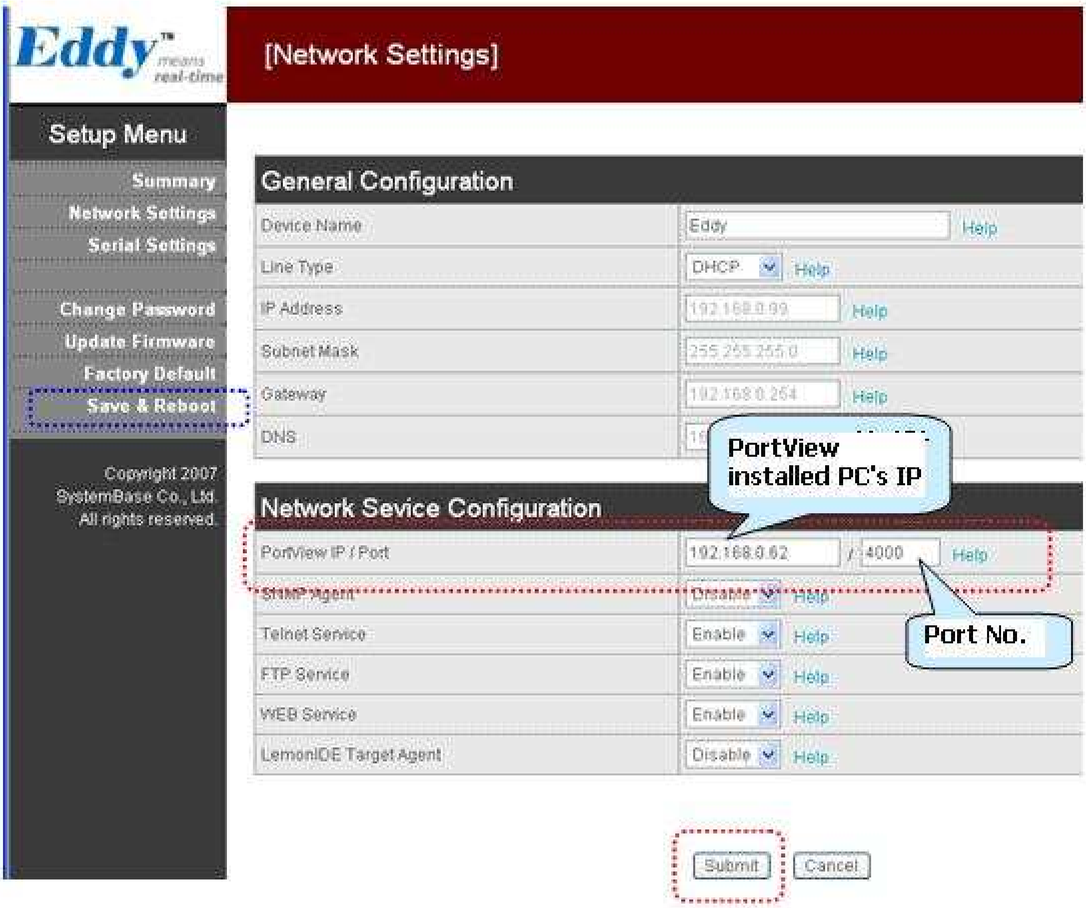
IP Configure: Gives temporary IP to the selected device..

Telnet: Connects to the selected device using Telnet.

Web: Connects to the selected device using Web.

2.2 Configuring Device Server Environment for Interaction with PortView

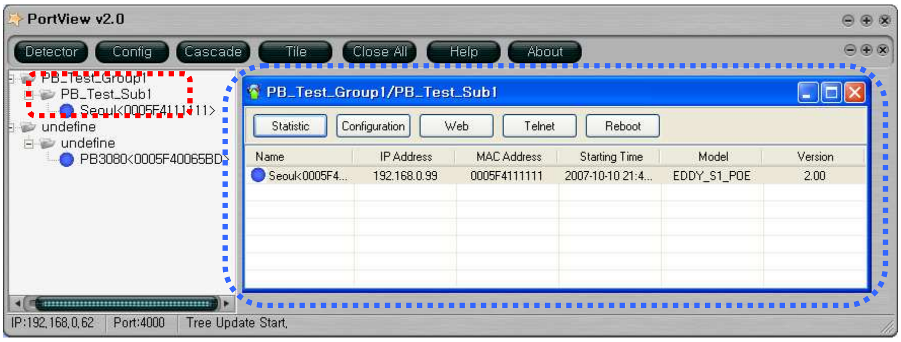
For PortView to interact with Eddy, Eddy must be setup with the IP of the PC in which the PortView is ran on and default port(4000) used on the PortView. First connect to Eddy using Web and change settings as shown below.



2.3 SubGroup Window

SubGroup window manages devices in the subgroup. When subgroup in the left window is selected by double clicking, the devices contained in the subgroup is shown in the device list window on the right

Statistic: Shows statistics of the selected device.



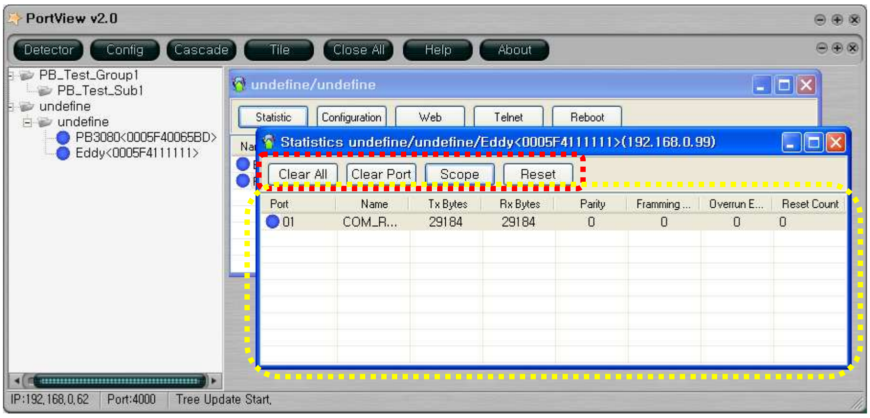
Configuration: Shows configuration of the selected devices.

Web: Executes Web Config window of the selected device.

Telnet: Opens Telnet window to the selected device.

Reboot: Resets selected device.

2.3.1 Statistic



Clear All: Clears numbers of all ports on statistics window such as Rx/TxByte

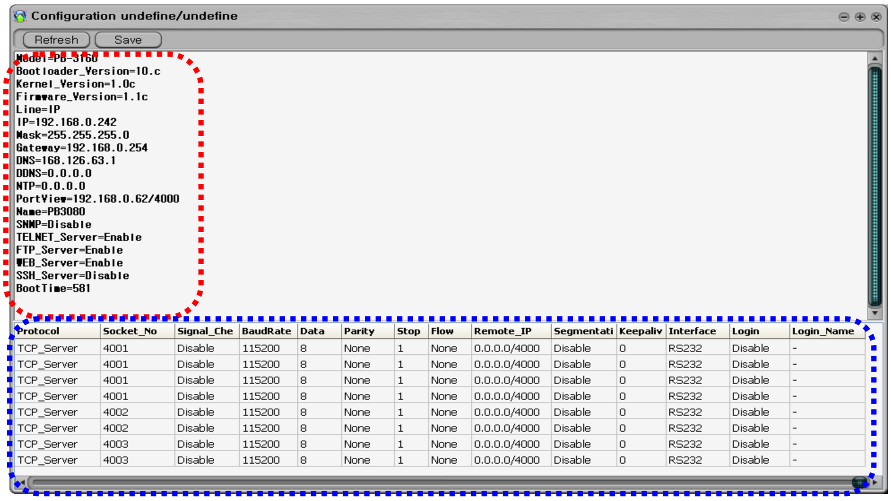
Clear Port: Clears numbers of selectd port on statistics window such as Rx/TxByte.

Scope: One scope can be executed per PortView. Provides scope ability to the DATA IO which runs independent of PortView.

Reset: Resets each port.

2.3.2 Configuration

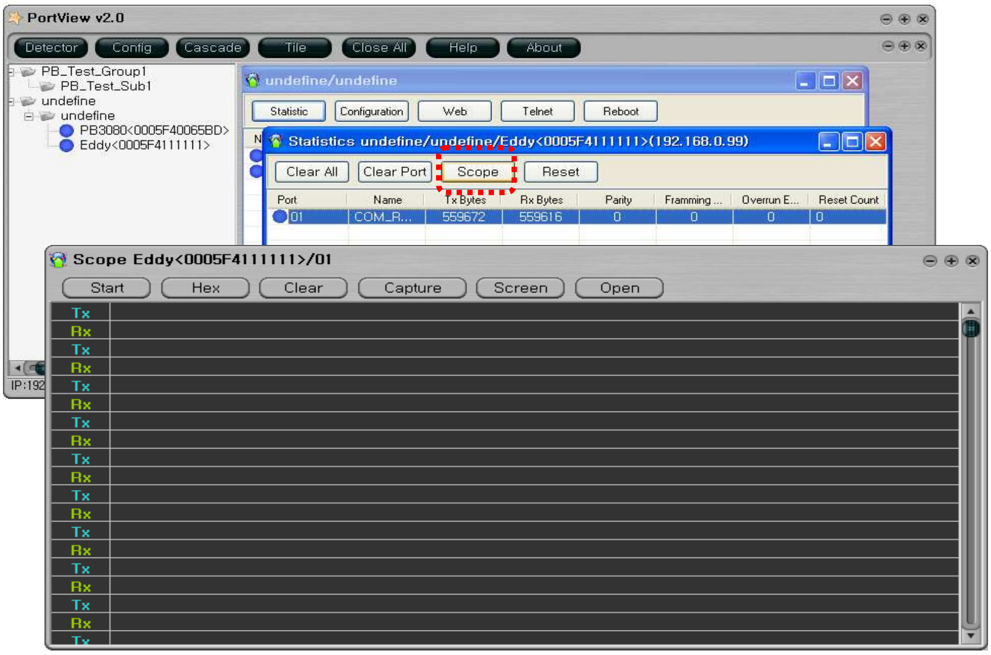
Shows configuration information of selected device.



2.3.3 Scope Ability

Scope Ability monitors Data IO. Input/Output can be monitored in Hex/ASCII mode.

Select the port you wish to examine, click “Scope” and the scope window will appear.



Start/Stop: Start or stop scoping.

Hex/ASCII: Selects data display type.

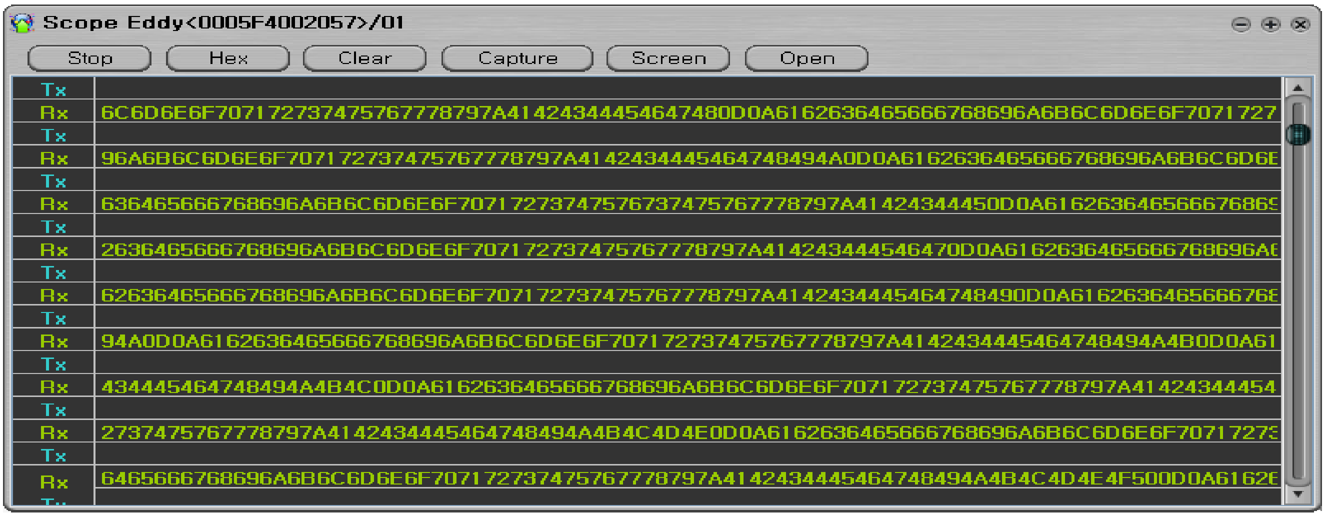
Clear: Clears printed values.

Capture: Captures printed values. Save after capture.

Screen: Selects background color and characteristic color.

Open: Opens captured & saved file.

[* Scoped Serial Port Data]

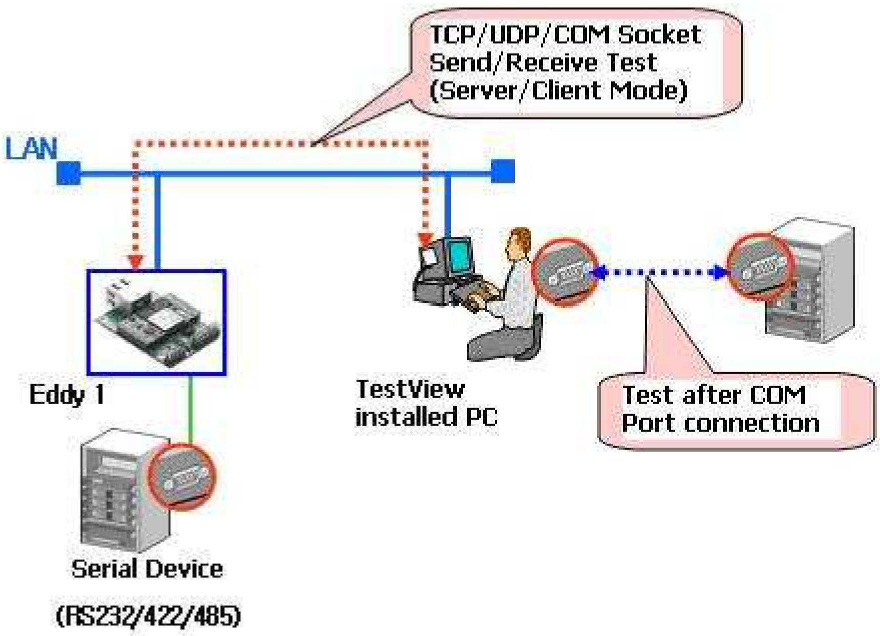


TCP/UDP/COM Testing Program

TestView

1. Summary

TestView is an application program for Windows for testing serial and socket communication that can easily and accurately inspect serial communication devices such as multiports, embedded modules and device servers. Provides TCP,UDP server/client ability and can inspect all serial communication related devices regardless of their manufacturer through burning test and performance test.



[*Understanding TestView]

Primary Functions of TestView

COM serial port open/test

TCP/UDP port open/test

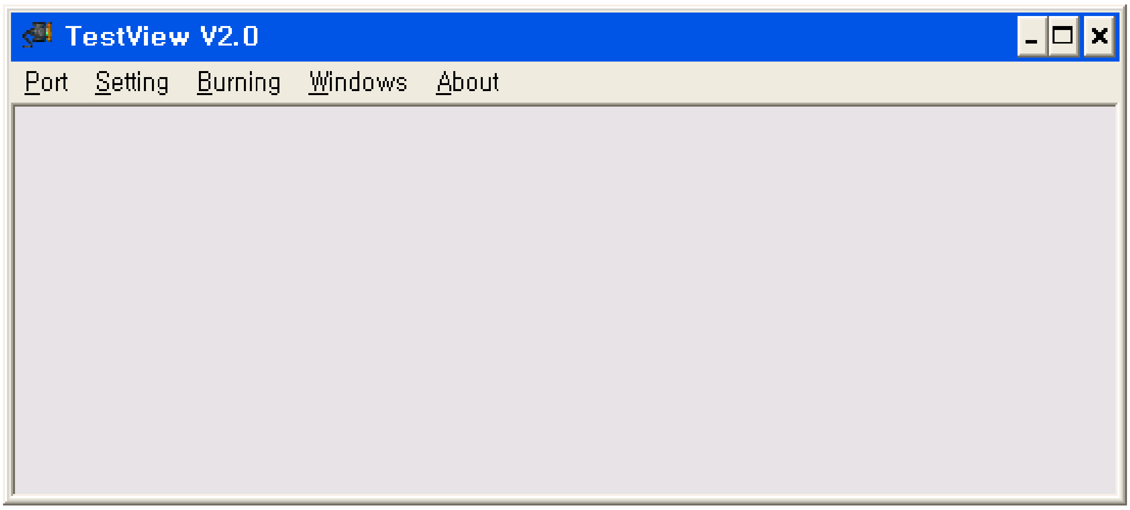
COM port burning test

TCP port burning test

Therefore, TestView is a communication test program that can examine actual COM port, virtual COM port, TCP connection port and UDP connection port with port open and data transmit/receive testing.

2. How to Use

Screen Components



[* TestView Screen]

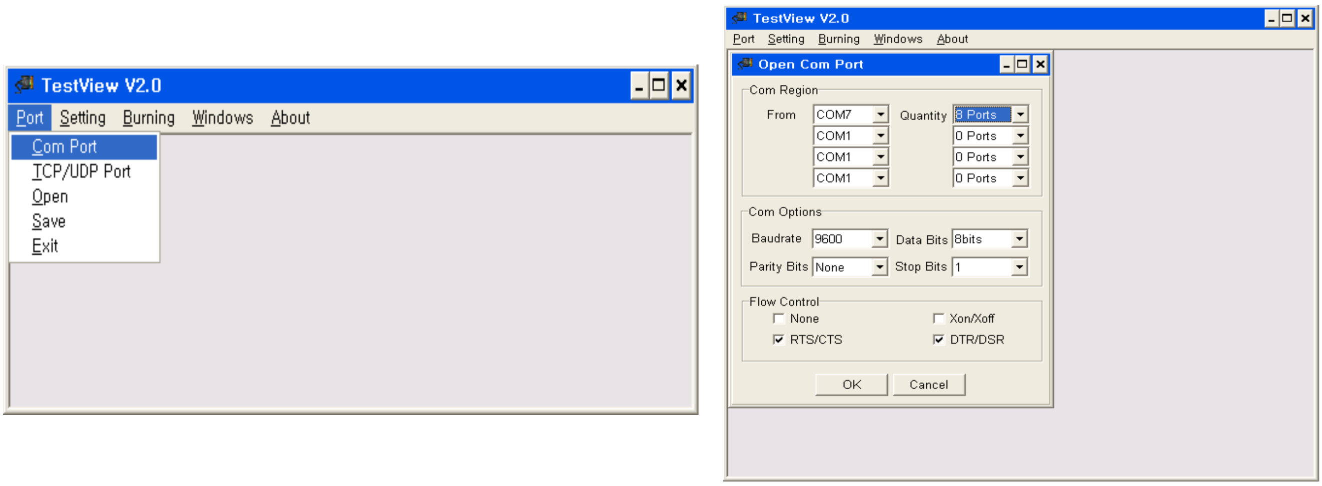
2.1 COM Port

Run test on COM Ports. It can

Check receiving data on COM Port on a new window.

Check throughput.

Transmit test data through COM port.



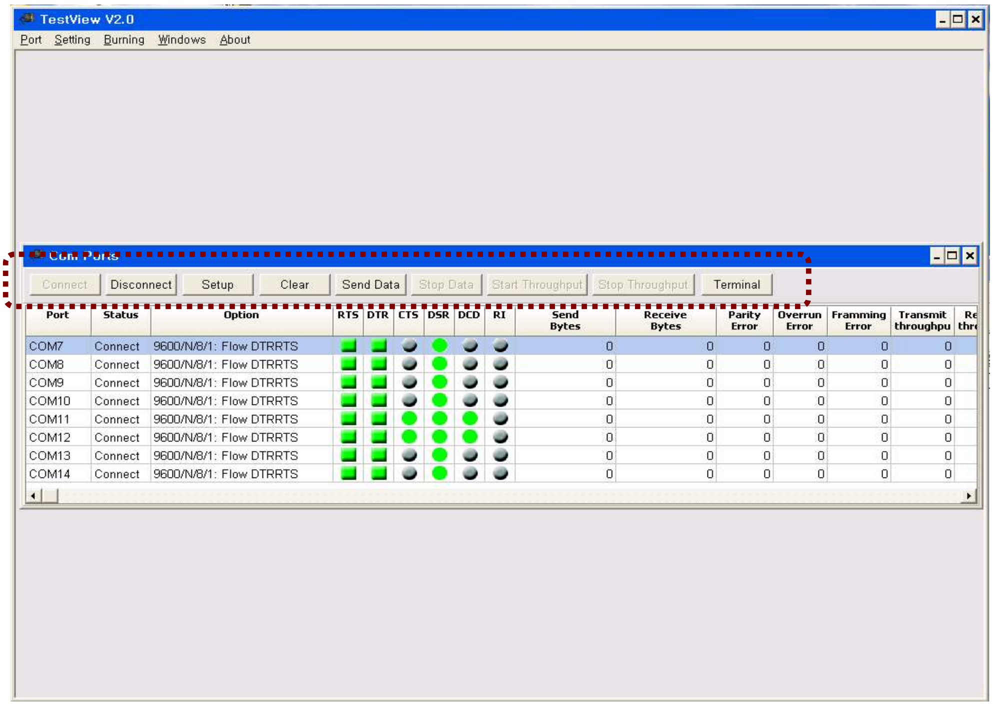
[* Opening a new window with “COM Port”]

Selecting initial COM port number and the number of ports will open ports. Also, specific options can be

assigned.

“COM Ports”window appears with 8 ports(COM7 ~ COM14) connected

Buttons on top will affect all ports on “COM Ports” window.



[* COM Port window opened]

Connect : Opens all selected ports.

Disconnect : Closes all selected ports connected.

Setup : Modifies initial communication settings. (Single port only)

Clear : Clears count values for all selected ports.

Send Data : Sends data A~Z to all selected ports.

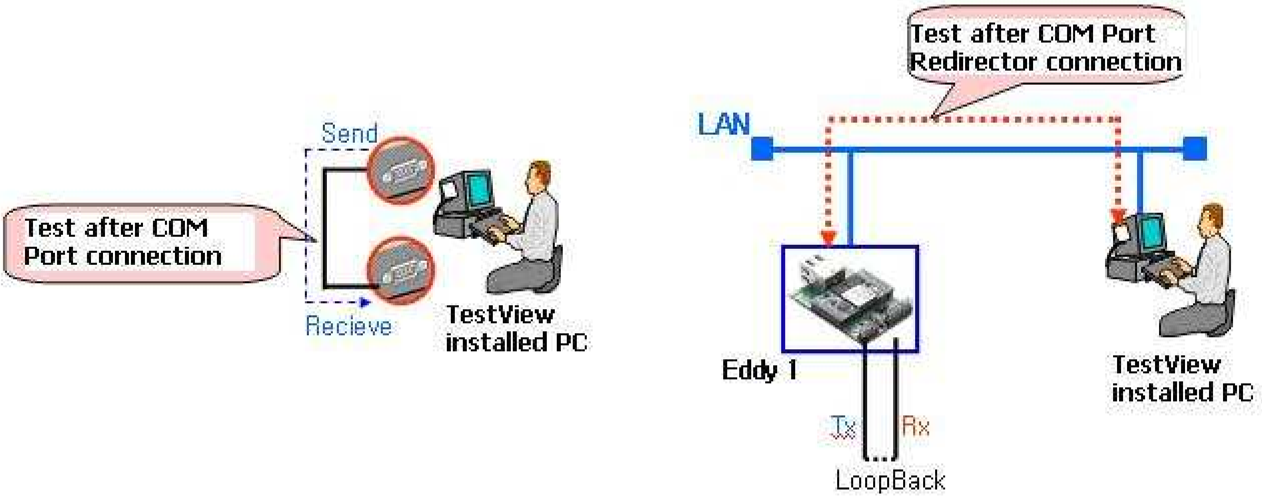
Start Throughput : Starts throughput for all selected ports.

Stop Throughput : Stops throughput for all selected ports.

Terminal : Runs emulator for all selected ports.

Open COM Port with Connect button as shown above.

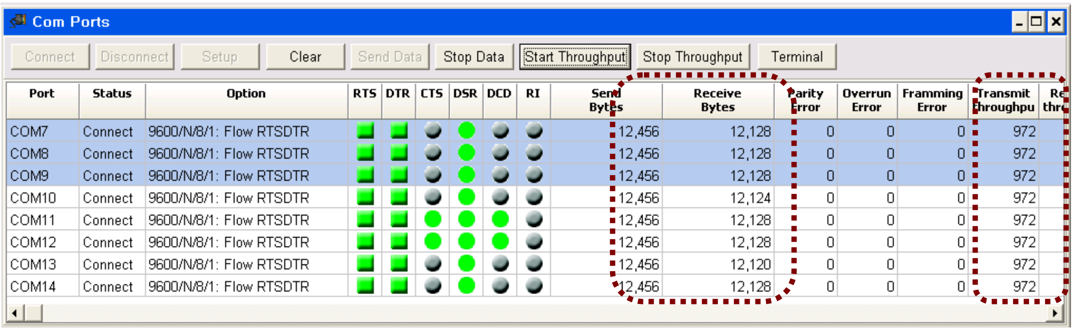
2.1.1 Send Data/Stop Data



[COM Port Test Methods]

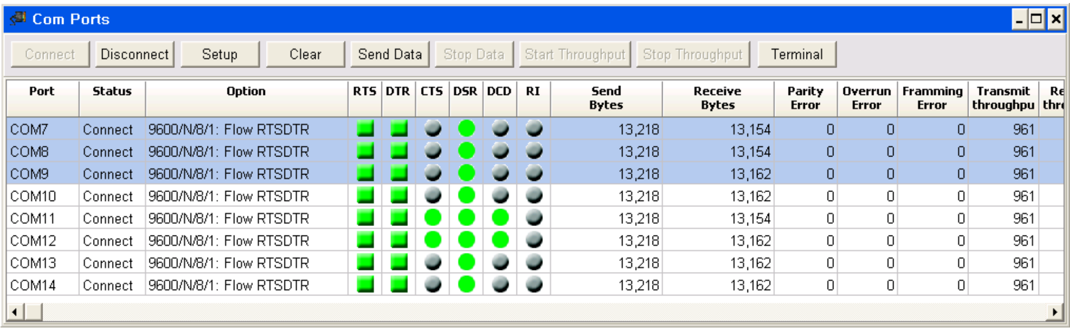
Transmit test data with current port.

Sending Data



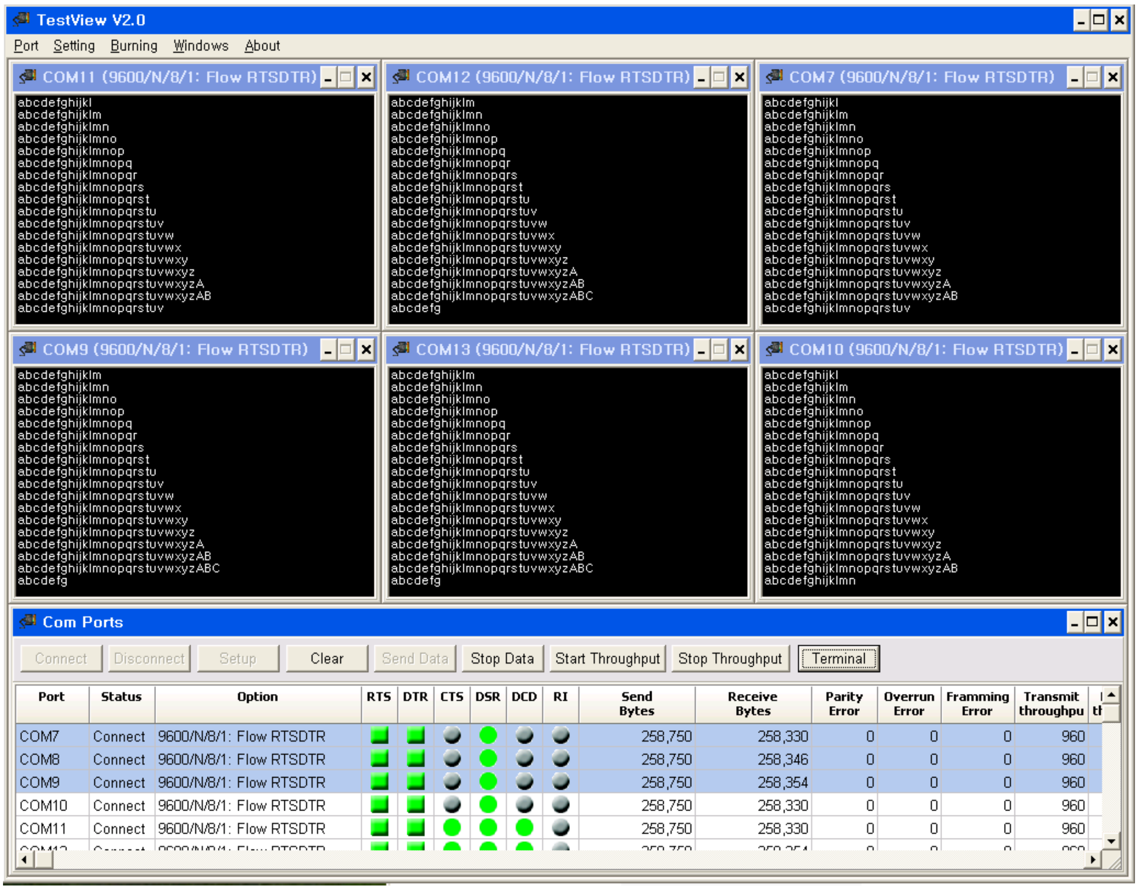
Port	Status	Option	RTS	DTR	CTS	DSR	DCD	RI	Send Bytes	Receive Bytes	Parity Error	Overrun Error	Framming Error	Transmit throughpu	Re thr
COM7	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	
COM8	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	
COM9	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	
COM10	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,124	0	0	0	972	
COM11	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	
COM12	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	
COM13	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,120	0	0	0	972	
COM14	Connect	9600/N/8/1: Flow RTSDTR							12,456	12,128	0	0	0	972	

Data sending stopped



Port	Status	Option	RTS	DTR	CTS	DSR	DCD	RI	Send Bytes	Receive Bytes	Parity Error	Overrun Error	Framming Error	Transmit throughpu	Re thr
COM7	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,154	0	0	0	961	
COM8	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,154	0	0	0	961	
COM9	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,162	0	0	0	961	
COM10	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,162	0	0	0	961	
COM11	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,154	0	0	0	961	
COM12	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,162	0	0	0	961	
COM13	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,162	0	0	0	961	
COM14	Connect	9600/N/8/1: Flow RTSDTR							13,218	13,162	0	0	0	961	

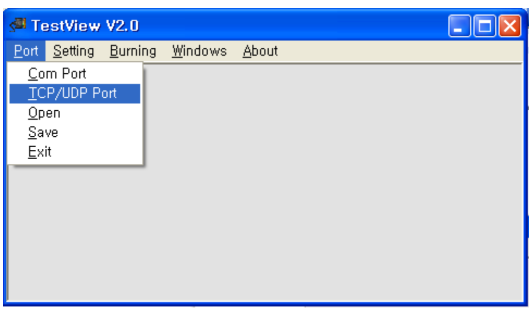
Data transmit/receive can be checked with Terminal window.

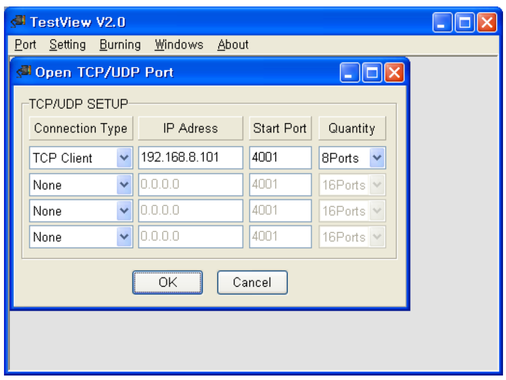


It can observe data received from COM Port. It is aligned in “Tile” with black background color.

2.2 TCP/UDP Port

Used to test with TCP Server/Client, UDP protocol.





Specify Connection Type, UDP protocol, IP Address, Start, Quantity settings and click OK.

Connection Type has 4 options.

TCP Client : Can connect to remote TCP server. Set TCP server’s IP and port to be connected.

TCP Server : User’s PC becomes a TCP server and operates. IP is set to PC’s IP.

UDP Client: Can connected to remote UDP server. Set UDP server's IP and port to be connected.

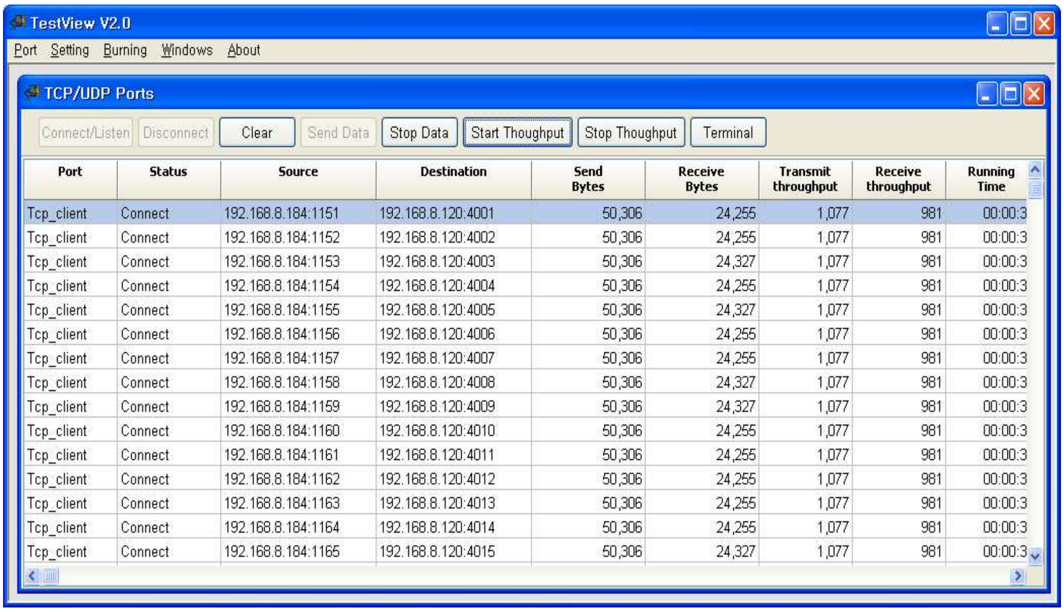
UDP Server : User's PC becomes a UDP server and operates. IP is set to PC's IP.

IP Address: Eddy's IP Address

Start Port: Selects initial port number.

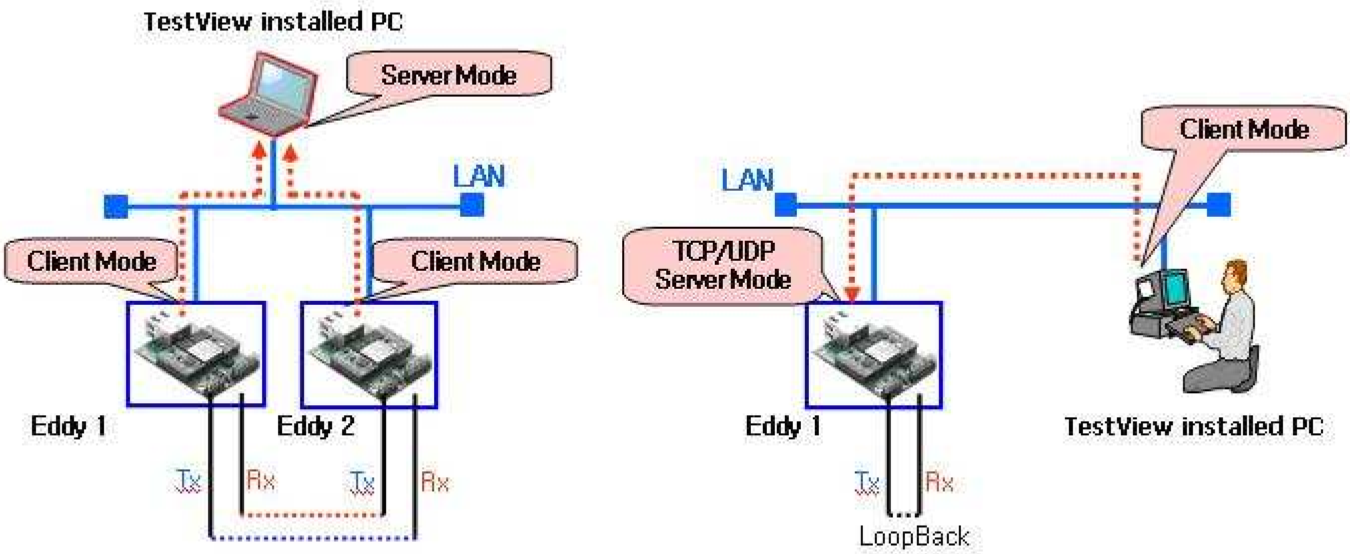
Quantity: Number of ports to be opened and tested.

After opening, testing procedure is same as testing in COM Port.



Port	Status	Source	Destination	Send Bytes	Receive Bytes	Transmit throughput	Receive throughput	Running Time
Tcp_client	Connect	192.168.8.184:1151	192.168.8.120:4001	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1152	192.168.8.120:4002	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1153	192.168.8.120:4003	50,306	24,327	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1154	192.168.8.120:4004	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1155	192.168.8.120:4005	50,306	24,327	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1156	192.168.8.120:4006	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1157	192.168.8.120:4007	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1158	192.168.8.120:4008	50,306	24,327	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1159	192.168.8.120:4009	50,306	24,327	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1160	192.168.8.120:4010	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1161	192.168.8.120:4011	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1162	192.168.8.120:4012	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1163	192.168.8.120:4013	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1164	192.168.8.120:4014	50,306	24,255	1,077	981	00:00:3
Tcp_client	Connect	192.168.8.184:1165	192.168.8.120:4015	50,306	24,327	1,077	981	00:00:3

[* TCP Client Port Screen]



[TCP/UDP Port Test Methods]

2.3 Burning

Burning test is needed for testing long time transmit/receive that can specify test time, transmit/receive counter, transmit/receive data amount which makes it suitable for product's aging test.

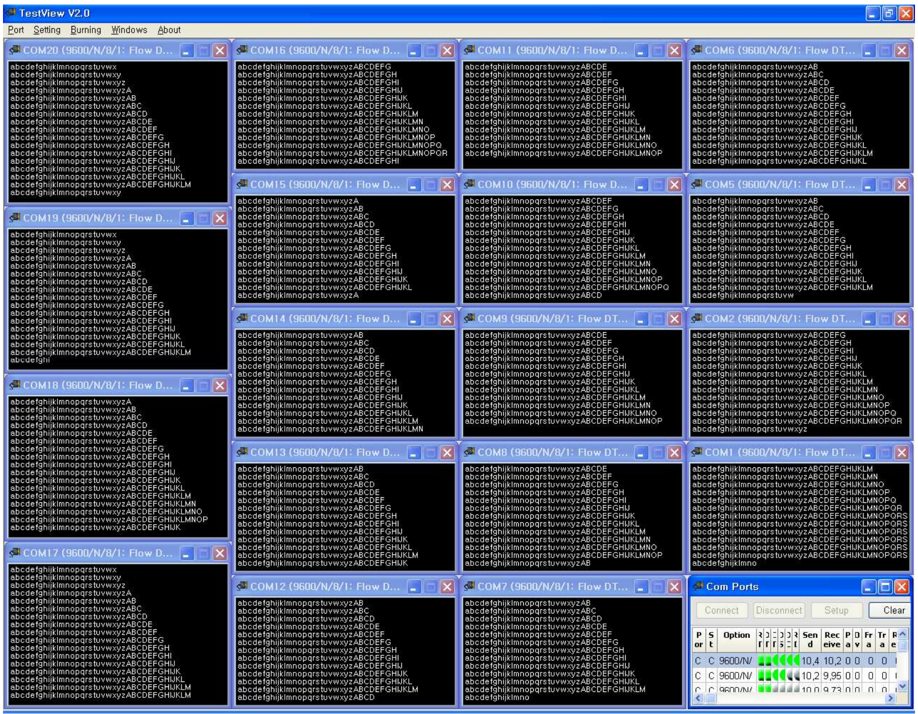


[*COM Port Burning example]

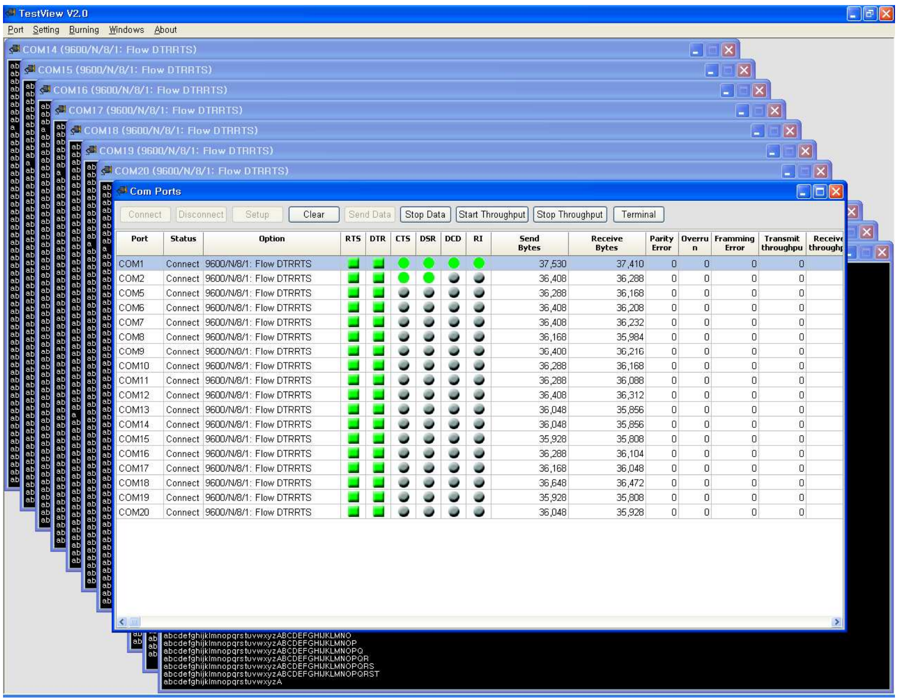
Number of errors can be counted during burning which allows users to measure product's performance.

2.3 Window Arrangement

Transmitted/received data in tested ports can be observed in numbers in TestView. Actual transmitted/received data can be observed through windows as well.



[* Windows Arranged in tiles]



[* Windows arranged in cascaded form]

Network Management

SNMP

1. Overview

Simple Network Management Protocol (SNMP) forms part of the internet protocol suite and is used in network management systems to easily monitor and control network-attached devices. Before the advent of SNMP, ICMP (Internet Control Message Protocol) was chiefly used to check host's status on the network by sending/receiving simple error messages – indicating, for example, that a host cannot be reached or that requested service is not available using "Echo request/reply" messaging functions of ICMP.

But as networks grew, and as Internet became prevalent, simple ICMP no longer was adequate for efficient network management. SGMP(Simple Gateway Monitoring Protocol), HIMS, CMIP(Common Management Information Protocol)/CMIS(Common management information service) emerged as alternative to replace ICMP but eventually SNMP, an advanced form of SGMP rooted as industry standard.

Almost all popular operating systems such as Unix, Linux and Windows include SNMP and its related tools in their network package, and TCP/IP network devices such as a router also support SNMP as one of their basic features.

2. Usage

Followings are typical SNMP usages.

* Network Architecture Management: Hierarchy architecture of hosts on the network can be retrieved.

* Performance Management: Statistical data required for analyzing each network segment's performance (traffic rate, error rate, processing time, response time, etc) can be retrieved

* Device Management: System information(CPU, memory, storage usage, etc) of each host on the network can be retrieved.

* Security Management: Provides function to control and protect relayed information. SNMP3 saw a drastic improvement on security management.

MIB

SNMP is a protocol for network management and SNMP itself does not define the information the

managed hosted has to offer. This information is defined in MIB(Management Information Base) and it describes the structure of management data of a device subsystem. This predefined structured information; MIB includes information of the system, network usage and network interface.

MIB is defined and managed by IANA(Internet Assigned Number Authority) and takes on form of a tree-like hierarchy for easy use and expansion. MIB can be amended usually by a device vendor to best reflect their product's characteristics. Such cases which expanded from IANA defined MIB is called expanded MIB.

MIB's versioned 1 & 2, notated as MIB-1 and MIB-2 is currently available. MIB-2 is an expanded version of MIB-1 and encompasses approximately 170 objects including all MIB-1's defined objects. Network objects are defined in forms of RFC. Eddy uses RFC-1212 and RFC1213 for network management and RFC-1659 for managing serial devices.

SNMP Manager/SNMP Agent

SNMP is a protocol and applications are required to collect network management data using SNMP. Generally, applications that utilize network protocols come in form of Server/Client model. This is also true for SNMP where terminology SNMP Manager/SNMP Agent is used rather than Server/Client.

A SNMP Agent is installed and runs on each managed system(network element or device) and reports the collected information via SNMP to information requesting managing systems (SNMP manager).

An SNMP Agent and SNMP Manager corresponds using three main tasks described below.

* GET (reads managed system's status) : SNMP Manager(Management system) requests GET to SNMP Agent (managed system), and SNMP Agent reports or exposes managed system's status in form of variable to requesting SNMP manager.

* SET (configures managed system) : SNMP Manager can send configuration updates or controlling requests to SNMP Agent using SET command.

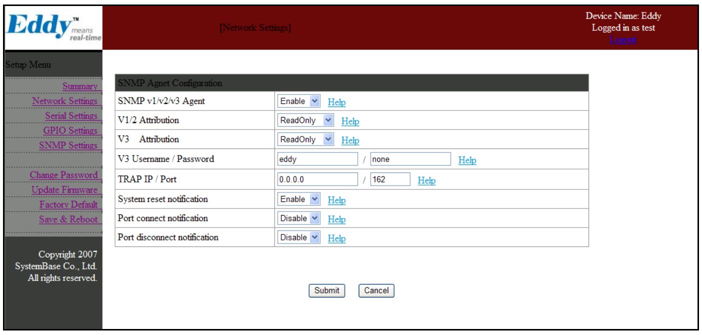
* TRAP (reports or alerts managed system's status without being asked by managing system) : SNMP Manager sends queries periodically to managed system using Polling method. When for some reason (usually due to network traffic) communication between SNMP Manager and Agent is delayed or is cut off, SNMP Manager will 'Retry' the non-responding SNMP agent. In such cases, there is a danger of Polling algorithm falling into an infinite loop and infinitely polling all managed systems listed in its polling list. To prevent such fallacy, SNMP agent may report its status using TRAP interrupt.

3. Using Eddy's SNMP

* Activating Eddy's SNMP

Eddy's SNMP encompasses RFC-1212, RFC-1213 and RFC-1659. RFC-1212 and RFC-1213 are standard MIB used for delivering network and system information and are supported by practically all network devices. RFC-1659 is standard MIB which various serial information, for example, communication line status and number of changing signal lines, are defined.
Besides the basic definitions on RFC-1659, environment setting information, usage and connection state of serial port, operating environment change, and reset function definitions have been added to Eddy's SNMP. A SNMP Manager can not only monitor and configure Eddy's operating environment status, but it can also modify operating settings in real time basis.

Eddy's SNMP Agent setting webpage and its setting values and explanation are shown below.



SNMP v1/v2/v3 Agent	Enable/Disables SNMP Agent
V1/2 Attribution	When using SNMP V1 and V2, sets read or write attribute of SNMP Agent. ReadOnly – SNMP agent can read only ReadWrite -- SNMP agent can read and write
V3 Attribution	When using SNMP V3, sets read or write attribute of SNMP Agent. ReadOnly – SNMP agent can read only ReadWrite -- SNMP agent can read and write.
V3 Username/ Password	When using SNMP V3, Sets Username and Password.
TRAP IP/ Port	Sets IP address and port number of the server that receives TRAP data.
System reset notification	When system resets, set to notify or not
Port connect notification	When connecting to a serial port, set to notify or not.
Port disconnect notification	When disconnecting to a serial port, set to notify or not.

* Setting MIB and using SNMP function on SNMP Manager.

The picture below shows an example of SNMP manager setup. Eddy has been selected as SNMP Agent, and MIBs have been registered. SNMP Manager program requested a GET to SNMP Agent, Eddy and Eddy reports the requested information to SNMP Manager. Below picture also depicts data scoping being processed due to a TRAP interrupt from SNMP Agent, Eddy.

[* Reference: MIB Browser Version 6.0 by iReasoning was used as SNMP manager program]

